2P−6

時間的要因を考慮したプロトコル検証法

畑 恵介, 金井 敦, 平川 豊
NTTソフトウェア研究所

1．はじめに

多数のノードから構成される現実のネットワークでは、輻輳や長時間保留などの時間的な要因で、設計時に予想できなかった障害が発生している。本研究では、これらの障害を設計段階で検出することを目的とし、入力としてメッセージシーケンスを用いて時間的な要因を考慮したプロトコル仕様の検証法を提案する。

2．対象とするモデル

モデル

仮定するシステムは、プロセスの集合によって構成される。また、それぞれのプロセスにはプロセス間通信チャネル（以下チャネルという）が、各プロセス対ごとに付随する。チャネルは無限大のバッファ容量を持つFIFOチャネルであると仮定する。プロセスが受信可能状態にない場合、着信メッセージはバッファ内で受信可能状態になるまで待つものとする。プロセスが受信可能状態であるが、メッセージが着信していない場合は、プロセスはそのメッセージの着信があるまで次のイベントを行なわない。

各プロセスでは、送受信イベントに伴う処理や、次のイベントを実行するまでの処理時間が必要である。このような各イベントの実行から、次のイベントが実行可能になるまでに必要な処理時間は、あらかじめ固定値として与えられているものとする。また、各チャネルの伝送遅延時間もあらかじめ固定値として与えられているものとする。

安全の定義

チャネルのキューが無限に増え続ける場合がシステムの最も危険な状態である。このためここでは第一歩として、極限状態であるキューの発散（キューが無限に増え続けること）について検討する。ここで、「キューが無限に増え続けるチャネルがシステムに存在しない場合」を「そのシステムは安全である」と言う。

3．検討の範囲

監視性

シーケンスは、ループ（陽に同一パターンを繰り返すようにシーケンスに記述されているものとする）で構成されている部分と、それ以外の部分にわけることができる。ループ以外の部分ではたとえキューが増え続けてもその数はたかだか有限個である。しかし、無限ループが存在した場合、発散する場合がある。従ってループのみを検討対象とする。

与えられたシステムにおいて、未完了な動作が無限におこらないように制限される性質として、「監視性」[1]という概念がある。監視性を有するシステムは、監視性の定義より「安全である」である。

まず、キューが発散する可能性のあるチャネルを検出するために、「局所的監視性プロセス群」というものを定義する。ここで、準備として任意のループで構成されている任意の隣接するプロセスの集合（部分システム）をとりだし、他のプロセスのシーケンスとチャネルを無視した部分的なシーケンスを考える。その部分的なシーケンスが監視性を満たす最大の部分システムを局所的監視性群（以下、群と呼ぶ）と呼ぶ。従って、「群は安全である」と言えるが、群と群の間のチャネルはキューが発散するおそれがある。従って、「群の集合からなる部分システムは安全でない可能性がある」と言える。

検討の対象

実際のシステムでは監視性がない場合でも問題なく動作していることがよくある。これは時間的な拘束条件により、「安全である」場合が存在するためである。

従って、群と群の間のチャネルに着目し時間要因を考慮して検証を行えば、そのシステムが安全であるか判定することができる。

今回は基本的な性質を明らかにするために、図1のようにある群（群Aとする）からもうひとつの群（群Bとする）に一方向に、一周期あたりにひとつのメッセージが送信される場合について、このチャネル内でキューが発散するかどうかの判定をするアルゴリズムの検討を行う。

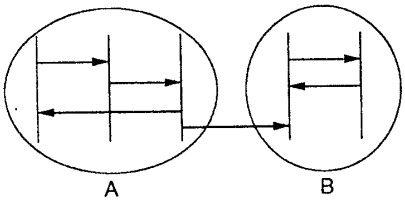

図1 検討の対象

4．アルゴリズム

支配時間グラフ

それぞれの群の周期を計算するために図2のような支配時間グラフを作成する。このグラフは有向グラフで、各プロセスに対応するノードを持つ。各プロセス（ノード）のメッセージシーケンスの先頭から送信方向にたどり、行くつくことができるプロセス（ノード）に対して枝を張る。各枝には支配処理時間が重みとして付される。支配処理時間とは、枝の始点のプロセスの先頭から終点のプロセスのループの終りへ行きつくパスの所要時間のうち最大の値である。また、各プロセスのループに入る以前の処理の遅れをΔとして考慮する。Δは各ノードごとに記述する。

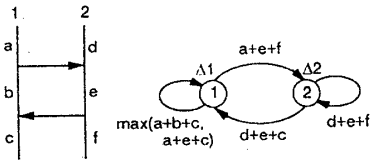

図2 支配時間グラフ

Verification of Time-Dependent Protocols
Keisuke HATA, Atsushi KANAI and Yutaka HIRAKAWA
NTT Software Laboratories

このグラフから、任意のプロセスがn周まわるのにかかる時間が計算できる。これはn本の枝を使って最後に着目しているノード（プロセス）にたどりつく遊歩道（経路）のうち、始点のノードのΔと各枝の重みを加算したものが最大になる遊歩道である。

定常状態の平均周期

定理１：どのような群でも必ず有限時間内で安定する平均周期が存在する。(証明略)

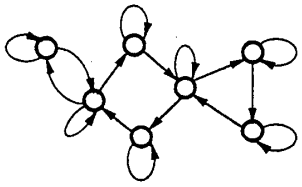

図3 グラフイメージ

群は監視性を有するので、支配時間グラフにおけるノードと枝の遊歩道は一般に複数の閉じたサイクルを構成する。十分大きな枝数x、すなわち、ループをx周回った時の処理時間を考える。グラフにおける枝数は有限であるから、この中で十分大きな枝数xをとる場合を考えると、サイクルを複数回まわることになる。また、全体の重みが最大になるように経路をとるので、平均の重みが最も重いサイクルを数多く回ることになる。この平均重みが最も重いサイクルの枝数をiとする。枝数$x+i$の時に全体の重みが最大になるよな遊歩道を選ぼうとすると、平均重みが最も重いサイクルをもう一回まわれば全体の重みが最も重くなる。すなわち、枝数が十分多くなるとこの平均重みが最も重いサイクルをまわり続けて枝数を増やす。これは十分大きな任意の枝数xについてすべて言える。この状態を安定状態と言う。また、この際の平均周期とは、そのサイクル全体の重みを枝数iで割ったものとなる。

キューの発散条件

定理２：安定状態における送信側の群Ａの平均周期T_Sと、受信側の群Ｂの平均周期T_Rとするとき、$T_S \geqq T_R$の関係がなりたてばこのチャネル間のキューは発散しない。(証明略)

それぞれの群Ａ、Ｂが安定状態でない時。群の周期が一定しないため、キューの数が増加する可能性がある。定理１より有限時間内ですべての群は安定状態に達する。しかし、この場合群Ａと群Ｂの間にチャネルが存在するため、送信側の群Ａの周期は安定するが、受信側の群Ｂはキューが溜っているかどうかで周期が安定しない。従って、安定状態において十分に多くのキューが溜っている場合を考える。この時、群Ｂは待ち時間なしにメッセージを受信できるため安定状態となる。もしキューが発散する場合には必ずこの状態を通過するので、この状態において安全であるといえれば、全ての状態について安全であると言える。

$T_S < T_R$の時は、群Ａのメッセージの送信間隔が群Ｂの受信間隔より速い。その結果、ループは無限回まわった時にキューは発散する。

$T_S > T_R$の時は群Ａのメッセージの送信間隔が群Ｂの受信間隔より遅いため、キューの数は減少してゆく。また、キューの数が減少してくると、受信側の群Ｂのメッセージの受信タイミングが不安定になり、再び増加する場合が考えられる。しかし、ある程度増加し群Ｂの周期が再び安定すると、キューの数は再び減少する方向に向かう。従って、この場合はキューは発散しない。

$T_S = T_R$の時は群Ａがメッセージを送信する周期と同じ周期で群Ｂが受信するため、キューは発散することはない。

検証方法

まず入力のメッセージシーケンスから支配時間グラフを作成する。次に群Ａと群Ｂの安定状態での平均周期をそれぞれ計算し、$T_S \leqq T_R$の関係が成り立てばキューは発散しない。

５．おわりに

考察

今回の検討の結果、安定した周期の状態になるまでの周回数が非常に大きな場合が存在することがわかっている。従って、TTG+[2]のような到達可能解析で安定状態の平均周期を求めるためには、安定状態に至るまで非常に多くのループを実行しなければならない。しかし、ここで提案した方法を用いれば、安定状態の平均周期を容易に計算することができる。

今後の課題

今回検討の対象とした群とチャネルの組合せは最も単純なものである。実際のシステムではこれ以上に複雑な組合せになるため、今後さらに検討が必要である。また、時間要因の入力情報として固定した処理の時間を与えたが、今後は処理時間について範囲を与えたものなどの検討が必要と思われる。今回はメッセージシーケンスチャートを入力として仮定し、これに時間要因を考慮して考察を行なった。しかし、状態遷移図を直接検証する方法や、状態遷移図からメッセージシーケンスに変換して検証する方法[3]が考えられるので、このような検証全体のフレームワークについても検討が必要である。

謝辞

本稿に関して有益な御意見並びに御協力をいただいたNTT基礎研究所の真鍋主任研究員、NTT情報通信網研究所の藤原主任研究員ならびにNTTソフトウェア研究所の市川グループリーダをはじめとするグループの皆様に心から感謝いたします。

参考文献

[1] 市川他: 並行処理の監視性と通信プロトコル実現への適用, 電子情報通信学会誌 B,Vol.J70-B,No.5,pp.565-575(1987).

[2] F.J.Lin *et al.*: On the Verification of Time-Dependent Protocols Using Timed Reachability Analysis, Proc.Hawaii Int.Conf.on Sys.Sci.,22nd,z,pp.285-294(1989).

[3] H.Saito *et al.*: Protocol Validation Tool and Its Applicability to Responsive Protocols, Proc.of 2nd IWRCS, pp98-103(1992).